

NASA/TM-1999-206570



A Parallel, Finite-Volume Algorithm for Large-Eddy Simulation of Turbulent Flows

*Trong T. Bui
Dryden Flight Research Center
Edwards, California*

National Aeronautics and
Space Administration

Dryden Flight Research Center
Edwards, California 93523-0273

January 1999

NOTICE

Use of trade names or names of manufacturers in this document does not constitute an official endorsement of such products or manufacturers, either expressed or implied, by the National Aeronautics and Space Administration.

Available from the following:

NASA Center for Aerospace Information (CASI)
7121 Standard Drive
Hanover, MD 21076-1320
(301) 621-0390

National Technical Information Service (NTIS)
5285 Port Royal Road
Springfield, VA 22161-2171
(703) 487-4650

A PARALLEL, FINITE-VOLUME ALGORITHM FOR LARGE-EDDY SIMULATION OF TURBULENT FLOWS

Trong T. Bui*
NASA Dryden Flight Research Center
Edwards, California

Abstract

A parallel, finite-volume algorithm has been developed for large-eddy simulation (LES) of compressible turbulent flows. This algorithm includes piecewise linear least-square reconstruction, trilinear finite-element interpolation, Roe flux-difference splitting, and second-order MacCormack time marching. Parallel implementation is done using the message-passing programming model. In this paper, the numerical algorithm is described. To validate the numerical method for turbulence simulation, LES of fully developed turbulent flow in a square duct is performed for a Reynolds number of 320 based on the average friction velocity and the hydraulic diameter of the duct. Direct numerical simulation (DNS) results are available for this test case, and the accuracy of this algorithm for turbulence simulations can be ascertained by comparing the LES solutions with the DNS results. The effects of grid resolution, upwind numerical dissipation, and subgrid-scale dissipation on the accuracy of the LES are examined. Comparison with DNS results shows that the standard Roe flux-difference splitting dissipation adversely affects the accuracy of the turbulence simulation. For accurate turbulence simulations, only 3–5 percent of the standard Roe flux-difference splitting dissipation is needed.

Nomenclature

A	area
$ \hat{A} $	Roe flux-difference splitting matrix
A_s	area of duct side walls
c	local speed of sound
C	SGS model constant

*Trong T. Bui, Aerospace Engineer, AIAA member, (805) 258-2645, e-mail: trong.bui@dfrc.nasa.gov.

Copyright © 1999 by the American Institute of Aeronautics and Astronautics, Inc. No copyright is asserted in the United States under Title 17, U.S. Code. The U.S. Government has a royalty-free license to exercise all rights under the copyright claimed herein for Governmental purposes. All other rights are reserved by the copyright owner.

CFD	computational fluid dynamics
CFL	Courant number
C_s	Smagorinsky constant
c_v	specific heat at constant volume
d	normal distance from a solid wall
d^+	normal distance from a solid wall in wall units, $d^+ = \frac{\rho u_\tau d}{\mu}$
D	entire flow domain
D_H	hydraulic diameter
DNS	direct numerical simulation
$d\mathfrak{s}$	elemental surface area on the boundary of a control volume
dv	elemental volume of a control volume
E_t	total energy/unit volume
\mathbf{f}	normal component of the inviscid flux vector
\mathbf{F}	flux vector of the Navier-Stokes equations
FDS	flux-difference splitting
\mathbf{F}_i	inviscid flux vector
\mathbf{F}_v	viscous flux vector
G	spatial filter used in the LES equations
I	total number of grid points in the streamwise direction
j	Jacobian determinant
J	total number of grid points in the wall-normal direction
k	conduction heat-transfer coefficient
K	total number of grid points in the spanwise direction
LES	large-eddy simulation
MPI	Message-Passing Interface

\hat{n}	normal unit vector
p	static pressure
P_g	mean pressure gradient
PVM	Parallel Virtual Machine
q^2	trace of the SGS Reynolds stress tensor
q_l	SGS term in the LES energy equation
R	specific gas constant
S	total area on the boundary of a control volume
SGS	subgrid scale
S_{kl}	velocity gradient tensor
t	time
T	temperature
u	x-component velocity
u_{ave}	mean streamwise velocity
$\frac{-\overline{u'v'}}{2}$	mean Reynolds stress
u_τ	friction velocity, $u_\tau = \sqrt{\frac{\tau_w}{\rho}}$
U	state vector of the Navier-Stokes equations
v	y-component velocity
V	total volume of a control volume
w	z-component velocity
x, y, z	coordinates of the physical space
δ_{kl}	Kronecker delta
Δ	width of filter used in LES equations
Δt_s	sampling time
ε_l	scaling factor for Roe flux-difference splitting
μ	molecular viscosity coefficient
ξ, η, ζ	coordinates of the computational space
ρ	static density
σ_{kl}	SGS term in the LES momentum equations (the SGS Reynolds stress tensor)
τ_{kl}	viscous stress tensor
τ_w	wall shear stress

Subscripts

a	node index
L	flow conditions to the left of a cell face
rms	root mean square
R	flow conditions to the right of a cell face

Superscript

n	time level
$-$	cell-averaged quantities in the Navier-Stokes equations, or filtered or large-scale quantities in the LES equations
\sim	Favre-filtered (density-weighted) variables
\sim	vector quantity

Introduction

Turbulence dominates the internal flows in aircraft jet engine components such as inlets, ducts, and nozzles and has been found to significantly influence engine noise and performance. Analytical tools are therefore needed to provide accurate predictions of these turbulent flows and allow engineers to explore the underlying flow physics, which would allow better aeropropulsion flow components to be designed and used in the aerospace industry.

Direct numerical simulation (DNS) of the turbulent flow inside of complete jet engines is presently not possible because of the tremendous computational resources required; however, technologies that potentially could make such a feat possible in the future are available today. These technologies include large-eddy simulation (LES) of turbulent flows, unstructured computational fluid dynamics (CFD) algorithms, and parallel computer systems. Large-eddy simulation has been shown to provide accurate turbulent flow simulation at a fraction of the cost of direct simulation. With unstructured CFD algorithms, complex three-dimensional aerodynamics shapes, including complete aircraft geometries, have been modeled using a single grid. Large-eddy simulation and unstructured CFD algorithms require large computing resources that potentially can be provided by the emerging parallel computer systems. By linking together hundreds or even thousands of individual processor nodes, the parallel computer systems can deliver significant advances in

computational resources in terms of memory, storage, and computing speed.

The above three technologies have been the subjects of ongoing intensive research, and a large body of knowledge has been separately accumulated on each of these subjects. The objective of this research is to develop a turbulence simulation tool using a combination of all of these technologies. The accuracy and efficiency of such a tool for turbulence simulations are then examined in detail from the LES of fully developed turbulent flow in a square duct.

Use of trade names or names of manufacturers in this document does not constitute an official endorsement of such products or manufacturers, either expressed or implied, by the National Aeronautics and Space Administration.

Numerical Algorithm

Development of the numerical algorithm has previously been described in detail.¹ This algorithm has previously been validated for time-accurate inviscid Euler simulations² and three-dimensional viscous Navier-Stokes simulations³ with good results. To describe the numerical algorithm, the Navier-Stokes equations are used in this section. These equations can be written in vector form as

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F} = 0 \quad (1)$$

where \mathbf{U} is the state vector and \mathbf{F} is the flux vector of the Navier-Stokes equations.

The above equation is discretized using the finite-volume approach. In this approach, equation (1) is integrated over a finite volume. Assuming the grid does not change with time and using the Gauss divergence theorem, the resulting equation is

$$\frac{d}{dt} \bar{\mathbf{U}} = -\frac{1}{V} \int_S \mathbf{F} \cdot d\mathbf{s} \quad (2)$$

where

$$\bar{\mathbf{U}} = \frac{1}{V} \int_V \mathbf{U} dv \quad (3)$$

and

$$\mathbf{F} \cdot d\mathbf{s} = \mathbf{F} \cdot \mathbf{n} ds \quad (4)$$

To numerically solve equation (2), the major steps of the solution procedure are reconstruction, flux computation, and evolution. This standard, finite-volume solution procedure has been used in previous works and has been described in detail by Barth.⁴ The steps are given below.

Step One: Reconstruction

For the first step, reconstruction, a cell-centered scheme is used. The piecewise linear, least-square reconstruction procedure used here is similar to those used by Barth⁴ and Coirier.⁵ Each of the five primitive variables ρ , u , v , w , and p is assumed to linearly vary within a finite volume as:

$$U(x, y, z) = \bar{U} + U_x(x - \bar{x}) + U_y(y - \bar{y}) + U_z(z - \bar{z}) \quad (5)$$

where U can be any of the above variables. The bars in equation (5) denote cell-averaged values as defined in equation (3). When used for high-speed compressible flow simulations, a gradient limiter is normally used in equation (5) to ensure that the reconstruction polynomial does not produce new extrema near a flow discontinuity such as a shock wave. In this paper, the gradient limiter is not used because the test case is low-Mach number turbulent flow in a square duct.

Following Coirier,⁵ the gradients U_x , U_y , and U_z in the target cell are computed using a least-square procedure that minimizes the sum of the squares of the differences between the values computed using the reconstruction polynomial from the target cell and the cell averages of the support set. For a three-dimensional hexahedron cell, the support set is the six neighboring cells that share their faces with the target cell. Algebraically, the minimization statement above can be expressed as:

$$\begin{bmatrix} a_1 & b_1 & b_2 \\ b_1 & a_2 & b_3 \\ b_2 & b_3 & a_3 \end{bmatrix} \begin{bmatrix} U_x \\ U_y \\ U_z \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} \quad (6)$$

$$\begin{aligned}
a_1 &= \sum_{i=1}^6 (\bar{x}_i - \bar{x}_0)^2 \\
a_2 &= \sum_{i=1}^6 (\bar{y}_i - \bar{y}_0)^2 \\
a_3 &= \sum_{i=1}^6 (\bar{z}_i - \bar{z}_0)^2 \\
b_1 &= \sum_{i=1}^6 (\bar{x}_i - \bar{x}_0)(\bar{y}_i - \bar{y}_0) \\
b_2 &= \sum_{i=1}^6 (\bar{x}_i - \bar{x}_0)(\bar{z}_i - \bar{z}_0) \\
b_3 &= \sum_{i=1}^6 (\bar{y}_i - \bar{y}_0)(\bar{z}_i - \bar{z}_0) \\
c_1 &= \sum_{i=1}^6 (\bar{U}_i - \bar{U}_0)(\bar{x}_i - \bar{x}_0) \\
c_2 &= \sum_{i=1}^6 (\bar{U}_i - \bar{U}_0)(\bar{y}_i - \bar{y}_0) \\
c_3 &= \sum_{i=1}^6 (\bar{U}_i - \bar{U}_0)(\bar{z}_i - \bar{z}_0)
\end{aligned} \tag{7}$$

where $i = 1-6$ denotes the neighboring cells, and $i = 0$ denotes the target cell.

Step Two: Flux Computation

With the piecewise linear reconstruction, the unknown variables are continuous and assumed to linearly vary within a finite volume. However, no guarantee exists that the variables will be continuous across adjacent finite volumes because a different polynomial is used in each finite volume. As a result, a flux formula is needed to compute a single flux at a finite-volume boundary using fluxes from the adjacent volumes. In the numerical solution of the Navier-Stokes equations, splitting the total flux vector into the inviscid flux vector and viscous flux vector is convenient:

$$\mathbf{F} \cdot \hat{n} = \mathbf{F}_i \cdot \hat{n} + \mathbf{F}_v \cdot \hat{n} \tag{8}$$

For the viscous flux, a simple arithmetic average is used. The normal component of the inviscid flux vector, $\mathbf{F}_i \cdot \hat{n}$, is approximated using the Roe flux-difference splitting (FDS) method without the entropy correction. The entropy correction is normally used to remove the nonphysical expansion shock at the sonic

transition point and is not needed here because of the low-Mach number test cases.

Define the normal component of the inviscid flux vector as

$$\mathbf{f} = \mathbf{F}_i \cdot \hat{n} \tag{9}$$

Then \mathbf{f} can be computed using the Roe FDS method as

$$\mathbf{f} = \frac{1}{2}(\mathbf{f}_L + \mathbf{f}_R) - \frac{1}{2}|\hat{A}|(\mathbf{U}_R - \mathbf{U}_L) \tag{10}$$

where the subscripts **L** and **R** denote the flow conditions to the left and right sides of the cell face.

Figure 1 shows the definitions used for the left and right states. Consider a cell, A, and its neighbor, B, sharing a common face, 1-2. When the flux across face 1-2 is computed for cell A, the left state (**L**) of the face 1-2 is on the side of cell A, and the right state (**R**) is on the side of cell B. This definition of the left and right states of a face is used because the face normal unit vector \hat{n} , which also serves as the locally one-dimensional coordinate system for the wave propagation across face 1-2, points from cell A to cell B. The **L** and **R** states are reversed when the flux is computed for cell B.

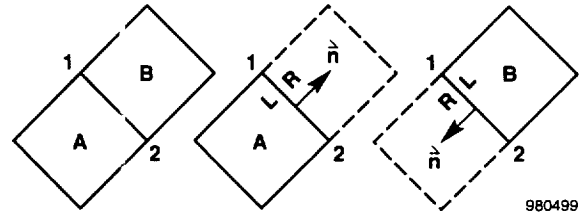


Figure 1. Definition of the left and right states of a face.

Step Three: Evolution

The two-stage, second-order, MacCormack time-marching algorithm is used to advance the solution in time. This explicit predictor-corrector time-marching method is accurate, efficient, and simple to implement on parallel computer systems.

Equation (2) can be rewritten as

$$\frac{d}{dt}\bar{U} = \mathbf{R} \tag{11}$$

When applied to equation (11), the MacCormack time-marching method gives

$$\bar{U}^{n+1} = \bar{U}^n + \Delta t \mathbf{R}^n \quad (12)$$

$$\bar{U}^{n+1} = \frac{1}{2} \left(\bar{U}^{n+1} + \bar{U}^n + \Delta t \mathbf{R}^{n+1} \right) \quad (13)$$

In addition to the major solution steps outlined above, describing how the volume and surface integrals in equations (2) and (3) are evaluated is important. Although the flow variables are approximated by discontinuous piecewise linear polynomials, the spatial coordinates x , y , and z of a finite volume are approximated by a continuous trilinear hexahedral element.⁶ This approach is the same that finite-element methods use to approximate the spatial coordinates. All of the integrations are then numerically evaluated using the one-point Gauss quadrature formula.

Each cell in the physical $\mathbf{x}(x, y, z)$ space is mapped to a trilinear hexahedral element in the $\xi(\xi, \eta, \zeta)$ space as shown in figure 2. The nodes, indexed 1 to 8, have the nodal coordinates in the ξ space shown in table 1.

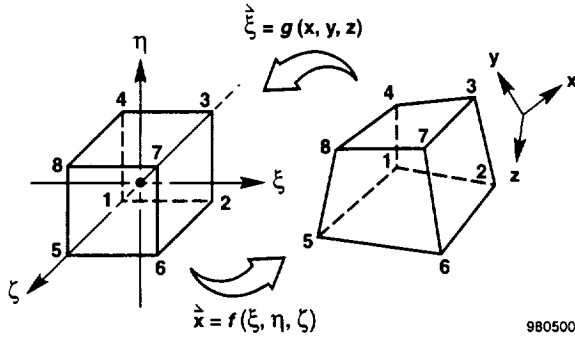


Figure 2. Local mapping between the physical finite-volume and the trilinear hexahedral element.

Table 1. Nodal coordinates in the ξ space.

Node index (a)	ξ_a	η_a	ζ_a
1	-1	-1	-1
2	1	-1	-1
3	1	1	-1
4	-1	1	-1
5	-1	-1	1
6	1	-1	1
7	1	1	1
8	-1	1	1

This type of mapping is different from the mapping used in generalized curvilinear finite-difference methods. In the finite-difference methods, the mapping applies to the entire computational block. In this algorithm, the local mapping applies to the local cell only. Each cell has its own mapping function, which is similar to a finite-difference multiblock method in which each cell is its own block. This ability gives considerable flexibility in the grid topology that can be used and is one of the strengths of this unstructured finite-volume algorithm.

The mapping from the ξ space to the \mathbf{x} space is given by:

$$\begin{aligned} x(\xi) &= \sum_{a=1}^8 N_a(\xi) x_a \\ y(\xi) &= \sum_{a=1}^8 N_a(\xi) y_a \\ z(\xi) &= \sum_{a=1}^8 N_a(\xi) z_a \end{aligned} \quad (14)$$

$$N_a(\xi) = \frac{1}{8} (1 + \xi_a \xi) (1 + \eta_a \eta) (1 + \zeta_a \zeta) \quad (15)$$

where the subscript a denotes the node index, ranging from 1 to 8. In the physical (x, y, z) coordinate system, node a has the coordinate (x_a, y_a, z_a) . In the computational (ξ, η, ζ) space, node a has the coordinate (ξ_a, η_a, ζ_a) . The coordinates (x_a, y_a, z_a) vary from cell to cell, depending on the physical grid. The coordinates (ξ_a, η_a, ζ_a) are the same for every cell and are shown in table 1.

To evaluate the volume integral, the following relation⁶ is used:

$$\begin{aligned} \int_V f(x, y, z) dv \\ = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 f(x(\xi, \eta, \zeta), y(\xi, \eta, \zeta), z(\xi, \eta, \zeta)) \\ j(\xi, \eta, \zeta) d\xi d\eta d\zeta \end{aligned} \quad (16)$$

where j is the Jacobian determinant, defined as

$$j = \det \left(\frac{\partial \mathbf{x}}{\partial \xi} \right) = \det \begin{bmatrix} x_\xi & x_\eta & x_\zeta \\ y_\xi & y_\eta & y_\zeta \\ z_\xi & z_\eta & z_\zeta \end{bmatrix} \quad (17)$$

The partial derivatives x_ξ , y_η , z_ζ , and so forth can be obtained by differentiating equation (14). Evaluating the determinant in equation (17) results in the following:

$$j = x_\zeta y_\xi z_\eta - x_\xi y_\zeta z_\eta - x_\zeta y_\eta z_\xi + x_\eta y_\zeta z_\xi + x_\xi y_\eta z_\zeta - x_\eta y_\xi z_\zeta \quad (18)$$

To evaluate the integral in equation (16), the one-point Gauss quadrature formula is used. In one dimension, the Gauss quadrature formulas are optimal, which means that accuracy of order $(2n)$ is achieved using (n) integration points. Gaussian rules for integrals in several dimensions are constructed by employing the one-dimensional Gaussian rules on each coordinate separately. In three dimensions, the one-point Gaussian rule is given as

$$\int_{-1}^1 \int_{-1}^1 \int_{-1}^1 f(\xi, \eta, \zeta) d\xi d\eta d\zeta = 8f(\xi = 0, \eta = 0, \zeta = 0) \quad (19)$$

Using the tools described above, the volume of the cell is computed as

$$V = \int_V 1 dv \quad (20)$$

$$V = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 j(\xi, \eta, \zeta) d\xi d\eta d\zeta \quad (21)$$

Using equation (19),

$$V = 8j(\xi = 0, \eta = 0, \zeta = 0) \quad (22)$$

so that the cell volume is approximately eight times the Jacobian determinant evaluated at the center of the cell $\xi = 0, \eta = 0, \zeta = 0$. Note that equation (22) contains two approximations: the physical coordinates (x, y, z) in the cell are approximated by equation (14), and the one-point Gaussian rule given by equation (19) is used for the numerical integration. Better approximation of the cell volume can be obtained using a higher-order approximation for the physical coordinates and a Gaussian rule with more points.

Computing the centroid of the cell also requires the evaluation of the volume integral. The coordinates of a cell centroid are given by $(\bar{x}, \bar{y}, \bar{z})$. The numerical approximation for the x coordinate of the centroid is

developed below. Approximations for the y and z coordinates are made exactly the same way.

$$\bar{x} = \frac{\int x dv}{V} \quad (23)$$

$$\bar{x} = \frac{\int_{-1}^1 \int_{-1}^1 \int_{-1}^1 x(\xi, \eta, \zeta) j(\xi, \eta, \zeta) d\xi d\eta d\zeta}{\int_{-1}^1 \int_{-1}^1 \int_{-1}^1 j(\xi, \eta, \zeta) d\xi d\eta d\zeta} \quad (24)$$

$$\bar{x} = \frac{8x(\xi = 0, \eta = 0, \zeta = 0)j(\xi = 0, \eta = 0, \zeta = 0)}{8j(\xi = 0, \eta = 0, \zeta = 0)} \quad (25)$$

or

$$\begin{aligned} \bar{x} &= x(\xi = 0, \eta = 0, \zeta = 0) \\ \bar{y} &= y(\xi = 0, \eta = 0, \zeta = 0) \\ \bar{z} &= z(\xi = 0, \eta = 0, \zeta = 0) \end{aligned} \quad (26)$$

From equation (14),

$$\begin{aligned} x(\xi = 0, \eta = 0, \zeta = 0) &= \frac{1}{8} \sum_{a=1}^8 x_a \\ y(\xi = 0, \eta = 0, \zeta = 0) &= \frac{1}{8} \sum_{a=1}^8 y_a \\ z(\xi = 0, \eta = 0, \zeta = 0) &= \frac{1}{8} \sum_{a=1}^8 z_a \end{aligned} \quad (27)$$

With the approximations in this algorithm, the coordinates of the cell centroid are simply the averages of the coordinates of the eight nodes defining the three-dimensional hexahedron cell.

The task of evaluating the surface integrals is described next. The surface integrals on the right side of equation (2) are of the form $\int_S \mathbf{F} \cdot d\mathbf{s}$. To develop the procedure, one face of the hexahedron shown in figure 2 is considered. The results for the other faces can be obtained using the same procedure.

Consider face 6-2-3-7 of the hexahedron in figure 2. Figure 3 shows this face redrawn for convenience. Because $\xi = 1.0$ for the nodes 6, 2, 3, 7 as well as any other point on this face, equation (14) reduces to the following:

$$\begin{aligned}
x &= \frac{1}{4} \sum_{a=6,2,3,7} (1 + \eta_a \eta)(1 + \zeta_a \zeta) x_a \\
y &= \frac{1}{4} \sum_{a=6,2,3,7} (1 + \eta_a \eta)(1 + \zeta_a \zeta) y_a \\
z &= \frac{1}{4} \sum_{a=6,2,3,7} (1 + \eta_a \eta)(1 + \zeta_a \zeta) z_a
\end{aligned} \quad (28)$$

so that x , y , and z are functions of η and ζ only.

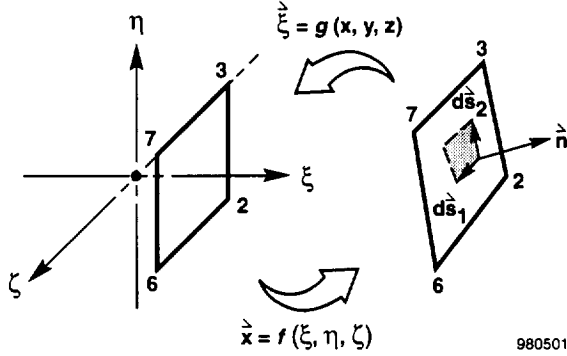


Figure 3. Coordinate systems for surface integral evaluation.

Following the development outlined in Greenberg,⁷ the tangent vectors $d\mathbf{s}_1$ and $d\mathbf{s}_2$ are tangents on the plane 6–2–3–7. $d\mathbf{s}_1$ is defined to be along the $\eta = \text{constant}$ curve on the face, and $d\mathbf{s}_2$ is along the $\zeta = \text{constant}$ curve. The vector $d\mathbf{s}_1$ may be expressed as follows:

$$\begin{aligned}
d\mathbf{s}_1 &= dx\hat{i} + dy\hat{j} + dz\hat{k} \\
dx &= x_\xi d\xi + x_\eta d\eta + x_\zeta d\zeta \\
dy &= y_\xi d\xi + y_\eta d\eta + y_\zeta d\zeta \\
dz &= z_\xi d\xi + z_\eta d\eta + z_\zeta d\zeta
\end{aligned} \quad (29)$$

where $d\xi = 0$ because the entire plane 6–2–3–7 is an ξ -constant plane, and $d\eta = 0$ because the vector $d\mathbf{s}_1$ is defined to be along the $\eta = \text{constant}$ curve. So

$$d\mathbf{s}_1 = (x_\zeta \hat{i} + y_\zeta \hat{j} + z_\zeta \hat{k}) d\zeta \quad (30)$$

and similarly

$$d\mathbf{s}_2 = (x_\eta \hat{i} + y_\eta \hat{j} + z_\eta \hat{k}) d\eta \quad (31)$$

The elemental area vector $d\mathbf{s}$, denoted by the shaded parallelogram, can be computed as

$$d\mathbf{s} = d\mathbf{s}_2 \times d\mathbf{s}_1 \quad (32)$$

or

$$\begin{aligned}
d\mathbf{s} &= (y_\eta z_\zeta - z_\eta y_\zeta) \hat{i} + (z_\eta x_\zeta - x_\eta z_\zeta) \hat{j} \\
&\quad + (x_\eta y_\zeta - y_\eta x_\zeta) \hat{k} \, d\eta d\zeta
\end{aligned} \quad (33)$$

Note that the order of the cross product in equation (32) is chosen so that the elemental area vector $d\mathbf{s}$ is positive pointing out of the cell and negative pointing into the cell. With the vector \mathbf{F} defined as

$$\mathbf{F} = F_x \hat{i} + F_y \hat{j} + F_z \hat{k} \quad (34)$$

the dot product is

$$\begin{aligned}
\mathbf{F} \cdot d\mathbf{s} &= [(y_\eta z_\zeta - z_\eta y_\zeta) F_x + (z_\eta x_\zeta - x_\eta z_\zeta) F_y \\
&\quad + (x_\eta y_\zeta - y_\eta x_\zeta) F_z] d\eta d\zeta
\end{aligned} \quad (35)$$

Finally, using the one-point Gaussian rule, the surface integral can be evaluated as

$$\begin{aligned}
\int_{(6237)} \mathbf{F} \cdot d\mathbf{s} &= 4[(y_\eta z_\zeta - z_\eta y_\zeta) F_x \\
&\quad + (z_\eta x_\zeta - x_\eta z_\zeta) F_y \\
&\quad + (x_\eta y_\zeta - y_\eta x_\zeta) F_z] \Big|_{\xi=1, \eta=0, \zeta=0}
\end{aligned} \quad (36)$$

The surface integrals for the other faces can be approximated in an analogous fashion. The results are given below.

$$\begin{aligned}
\int_{(1584)} \mathbf{F} \cdot d\mathbf{s} &= 4[(z_\eta y_\zeta - y_\eta z_\zeta) F_x \\
&\quad + (x_\eta z_\zeta - z_\eta x_\zeta) F_y \\
&\quad + (y_\eta x_\zeta - x_\eta y_\zeta) F_z] \Big|_{\xi=-1, \eta=0, \zeta=0}
\end{aligned} \quad (37)$$

$$\begin{aligned}
\int_{(8734)} \mathbf{F} \cdot d\mathbf{s} &= 4[(z_\xi y_\zeta - y_\xi z_\zeta) F_x \\
&\quad + (x_\xi z_\zeta - z_\xi x_\zeta) F_y \\
&\quad + (y_\xi x_\zeta - x_\xi y_\zeta) F_z] \Big|_{\xi=0, \eta=1, \zeta=0}
\end{aligned} \quad (38)$$

$$\begin{aligned}
\int_{(1265)} \mathbf{F} \cdot d\mathbf{s} &= 4[(y_\xi z_\zeta - z_\xi y_\zeta) F_x \\
&\quad + (z_\xi x_\zeta - x_\xi z_\zeta) F_y \\
&\quad + (x_\xi y_\zeta - y_\xi x_\zeta) F_z] \Big|_{\xi=0, \eta=-1, \zeta=0}
\end{aligned} \quad (39)$$

$$\begin{aligned}
(5678) \quad \int \mathbf{F} \cdot d\mathbf{s} &= 4[(y_\xi z_\eta - z_\xi y_\eta) \mathbf{F}_x \\
&+ (z_\xi x_\eta - x_\xi z_\eta) \mathbf{F}_y \\
&+ (x_\xi y_\eta - y_\xi x_\eta) \mathbf{F}_z] \Big|_{\xi=0, \eta=0, \zeta=1}
\end{aligned} \quad (40)$$

$$\begin{aligned}
(2143) \quad \int \mathbf{F} \cdot d\mathbf{s} &= 4[(z_\xi y_\eta - y_\xi z_\eta) \mathbf{F}_x \\
&+ (x_\xi z_\eta - z_\xi x_\eta) \mathbf{F}_y \\
&+ (y_\xi x_\eta - x_\xi y_\eta) \mathbf{F}_z] \Big|_{\xi=0, \eta=0, \zeta=-1}
\end{aligned} \quad (41)$$

Symbolically, the surface integrals of the opposite faces are the negative of each other (for example, face $\xi = 1$ as given by equation (36) and face $\xi = -1$ as given by equation (37)). However, for actual numerical values, each of the integrals will need to be separately evaluated because the integrands depend on the coordinates of the nodal points (x_a, y_a, z_a) and the Gaussian point coordinates.

To calculate the Jacobian determinant of the coordinate transformation, equations (14) and (15) are used. For example, the derivative x_ξ can be evaluated as follows:

$$\begin{aligned}
x(\xi) &= \sum_{a=1}^8 N_a(\xi) x_a \\
x_\xi &= \sum_{a=1}^8 N_{a,\xi} x_a
\end{aligned} \quad (42)$$

Governing Equations for Large-Eddy Simulation

In LES, the large scale of turbulence is computed directly in the numerical simulation, and the effects of the small scale stresses are modeled using a subgrid-scale (SGS) model. The governing equations for LES of turbulent flows can be obtained from filtering (local volume-averaging) the compressible Navier-Stokes equations. From Moin et al.,⁸ the LES equations for compressible flows (using tensor notation) are given by the following:

$$\frac{\partial}{\partial t} \bar{\rho} + \frac{\partial}{\partial x_k} \bar{\rho} \tilde{u}_k = 0 \quad (43)$$

$$\frac{\partial}{\partial t} \bar{\rho} \tilde{u}_k + \frac{\partial}{\partial x_l} \bar{\rho} \tilde{u}_k \tilde{u}_l + \frac{\partial}{\partial x_k} \bar{p} - \frac{\partial \tilde{\tau}_{kl}}{\partial x_l} + \frac{\partial \sigma_{kl}}{\partial x_l} = 0 \quad (44)$$

$$\begin{aligned}
&\frac{\partial \tilde{E}_t}{\partial t} + \frac{\partial}{\partial x_l} \tilde{E}_t \tilde{u}_l + \frac{\partial}{\partial x_l} \bar{p} \tilde{u}_l - \frac{\partial}{\partial x_l} \tilde{u}_k \tilde{\tau}_{kl} \\
&- \frac{\partial}{\partial x_l} \left(\tilde{k} \frac{\partial}{\partial x_l} \tilde{T} \right) + \tilde{u}_k \frac{\partial \sigma_{kl}}{\partial x_l} + c_v \frac{\partial q_l}{\partial x_l} = 0
\end{aligned} \quad (45)$$

The bar in the LES equations (43) to (45) denotes a filtered or large-scale flow quantity, defined as

$$\bar{f} = \int_D G(\mathbf{x} - \mathbf{x}') f(\mathbf{x}') d\mathbf{x}'$$

where G is a spatial filter and the integral is over the flow domain, D . The tilde in the LES equations denotes a Favre-filtered (density-weighted) variable, defined as

$$\tilde{f} = \frac{\bar{\rho} f}{\bar{\rho}}$$

The filtered ideal gas equation of state is given by

$$\bar{p} = \bar{\rho} R \tilde{T}$$

Moin et al.⁸ used an internal energy equation in their derivation of the LES equations. The LES total energy equation, equation (45), is obtained from adding the dot product of the LES momentum equation, equation (44), and the filtered velocity field \tilde{u}_k to the LES internal energy equation in Moin et al.⁸

The LES equations given by equations (43) to (45) are essentially the Navier-Stokes equations written for the filtered variables plus the additional subgrid terms in the momentum and total energy equations. Thus, the numerical algorithm developed in the last section can be used to solve the LES equations. The treatment of the subgrid terms are discussed in the next section.

Subgrid-Scale Models

Detailed studies have previously been performed to assess the relative importance of the subgrid terms in the filtered total energy equation for compressible turbulent shear flows at different Mach numbers.^{9, 10} These studies led to the conclusion that the energy subgrid terms may be neglected if the Mach number of the simulation is low. Because of the low Mach number of the turbulent square duct test case, this assumption is used.

The subgrid term in the momentum equations, equation (44), is

$$\sigma_{kl} = \bar{\rho}(u_k \tilde{u}_l - \tilde{u}_k \tilde{u}_l)$$

To close the system of LES equations, this term needs to be modeled. In Moin et al.,⁸ this term is approximated as follows:

$$\sigma_{kl} = -2C\bar{\rho}\Delta^2|\tilde{S}| \left(\tilde{S}_{kl} - \frac{1}{3}\tilde{S}_{mm}\delta_{kl} \right) + \frac{1}{3}q^2\delta_{kl} \quad (46)$$

where

$$q^2 = \sigma_{ii}$$

is the trace of the SGS Reynolds stress tensor. The filtered velocity gradient tensor is

$$\tilde{S}_{kl} = \frac{1}{2} \left(\frac{\partial \tilde{u}_k}{\partial x_l} + \frac{\partial \tilde{u}_l}{\partial x_k} \right)$$

and

$$|\tilde{S}| = (2\tilde{S}_{kl}\tilde{S}_{kl})^{\frac{1}{2}}$$

In equation (46), C is a constant to be determined according to the particular SGS model used. For LES of turbulent channel and duct flows using the Smagorinsky SGS model,¹¹ a value of $C = 0.01$ is commonly used with good results. Note that the constant C in equation (39) is the square of the Smagorinsky constant $C_s = 0.1$.

Unlike LES of isotropic turbulence, C is not constant in wall-bounded flows and varies according to distance from the wall. The dynamic SGS model developed by Germano et al.¹² would correctly determine the value for C using a dynamic procedure; however, this model is computationally expensive because of the extra filtering operations that must be done. Also, a question currently exists on the mathematical well-posedness of this model.¹³ Finally, the dynamic model has been known to compensate for the effects of numerical dissipation by automatically varying the magnitude of the constant C .

One of the main objectives of this research is to quantify the effects of the upwind numerical dissipation on the accuracy of the turbulence simulations. Also, the simpler Smagorinsky model has been found to work as well as the dynamic SGS model for this simple test case.¹⁴ As the result, the Smagorinsky SGS model is used in this study with the constant C given by the following:

$$C = 0.01 \left(1.0 - \exp \left(- \left(\frac{d^+}{25} \right)^3 \right) \right) \quad (47)$$

where d^+ is the normal distance from the wall in wall units, defined as

$$d^+ = \frac{\rho u_\tau d}{\mu}$$

and the friction velocity is defined as

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}}$$

Because the turbulent flow in the corner of the square duct encounters walls in two different directions, d is taken to be

$$d = \frac{2yz}{y + z + (y^2 + z^2)^{\frac{1}{2}}} \quad (48)$$

Equation (48) is frequently used in the turbulence modeling of flows in the vicinity of a wall corner. The variables y and z are the normal distances to the nearest walls in the y and z directions. Note that d tends to y as (y/z) tends to 0, and d tends to z as (z/y) tends to 0, which are the intended results.

In LES, the width of the filter used in the process of volume-averaging the Navier-Stokes equations, Δ , is typically chosen to be the grid spacing size. This study defined the grid spacing size to be $\sqrt[3]{V}$, where V is the cell volume.

The term q^2 in equation (46) is the isotropic part of the SGS Reynolds stress tensor. Like the rest of this tensor, the term cannot be calculated directly in an LES and has to be modeled. A number of different models for q^2 has been proposed.¹⁵⁻¹⁷ However, results from recent studies indicated that this term is not important for accurate LES of low-Mach number, low-Reynolds number compressible turbulent flows.

An evidence in support of the above conclusion was presented by Squires,¹⁸ who compared two different models of q^2 in addition to setting $q^2 = 0$. Squires found essentially no difference in the results of LES of compressible isotropic turbulence at a low Mach number and, in fact, observed that neglecting q^2 slightly improved the agreement between the LES and DNS results.

Vreman et al.⁹ confirmed the above results with their simulations of a three-dimensional temporal compressible mixing layer at a mean convective Mach number of 0.2. In *a priori* tests, the SGS model that neglects q^2 was found to give a better correlation with the DNS results. Furthermore, LESes conducted with the dynamic SGS model for q^2 were unstable for the cases that were studied.

For low-Mach number turbulence LES, neglecting the term q^2 will not introduce large errors in the results and is actually desirable in some cases, as the above findings showed. As a result, the term q^2 is neglected in this study. This assumption is analogous to the Stokes assumption for the viscous stress tensor in the Navier-Stokes equations. With the term q^2 omitted, the SGS stresses can be included in the Navier-Stokes equations by simply replacing the laminar viscosity coefficient μ with μ_{eff} where $\mu_{eff} = \mu + \mu_{SGS}$ and $\mu_{SGS} = C\bar{\rho}\Delta^2|\bar{S}|$.

Large-Eddy Simulation of Turbulent Flow in a Square Duct

To validate the numerical method for turbulence simulations of duct flows, LES of fully developed turbulent flow in a square duct is performed. For the purpose of comparison, a low-Reynolds number, square duct DNS solution is available. This DNS database was used by Mompean et al.¹⁹ to evaluate nonlinear k- ϵ turbulence models. Another DNS solution of the fully developed turbulent square duct flow at a slightly lower Reynolds number is also available.²⁰

Figure 4 shows the coordinate system and geometry for the square duct flow. In this test case, the Reynolds number based on the mean streamwise velocity and hydraulic diameter is 4800. Based on the friction velocity and hydraulic diameter, the Reynolds number is 320.

Table 2 shows a summary of the flow properties for the test case, assuming an average Mach number of 0.3 and standard sea level properties for air. The computational domain size used in the LES is $12D_H \times D_H \times D_H$. In choosing the size of the computational domain, care must be taken to ensure that the length of the computation domain is large enough to adequately contain the largest turbulence structure. Two-point velocity correlations for three different cross-stream positions in the duct were computed from the DNS solution by Gavrilakis.²⁰ The correlations for all three velocity components become essentially 0 at a

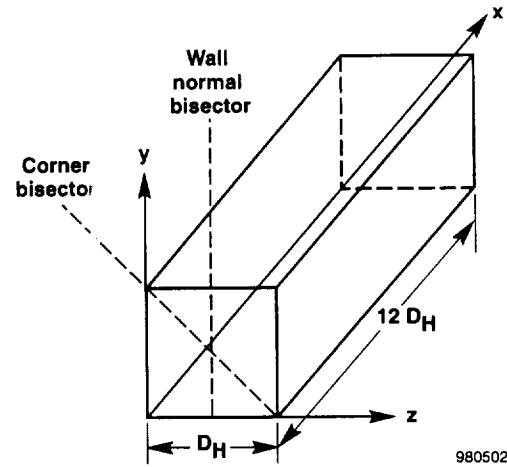


Figure 4. Coordinate system and geometry for the square duct flow.

duct length of approximately $6D_H$, so that a length of $12D_H$ should be adequate to capture the streamwise turbulence structures. Two different grids are used in the present LES, and table 3 shows the simulation parameters. The sampling time for the turbulent statistics is large compared to the time step size, but is small compared to the eddy turnover time in order to capture the unsteadiness of the turbulent flow.

Table 2. Flow properties for the turbulent square duct test case.

Flow properties	Values
Average Mach number, $\frac{u_{ave}}{c}$	0.3
Average streamwise velocity, u_{ave}	102.4 m/sec
Average friction velocity, u_τ	6.83 m/sec
$Re = \frac{\rho u_{ave} D_H}{\mu}$	4800
$Re_\tau = \frac{\rho u_\tau D_H}{\mu}$	320
Hydraulic diameter, D_H	7.37×10^{-4} m
Mean pressure gradient, P_g	-289,646 Pa/m
Eddy turnover time, $\frac{0.5 D_H}{u_\tau}$	5.4×10^{-5} sec

Table 3. Parameters for the LES

Parameters	Grid A	Grid B
Grid size (I × J × K)	129 × 90 × 90	257 × 100 × 100
Number of cells	1,013,888	2,509,056
Minimum resolution (in wall units)	30 × 1.88 × 1.88	15 × 1.69 × 1.69
Maximum resolution (in wall units)	30 × 4.86 × 4.86	15 × 4.37 × 4.37
Sampling time, Δt_s (sec)	6.0×10^{-7}	5.0×10^{-7}
Time step size, Δt (sec)	3.0×10^{-9}	2.5×10^{-9}
CFL number	0.98	0.93

The periodic boundary condition used for the inflow and exit boundary of the square duct is similar to the one used by Coleman et al.²¹ With this boundary condition, all of the flow conditions are periodic at the duct inlet and exit planes. The driving pressure gradient in the duct is specified in the flow equations as an extra body force term.

To reduce the number of iterations required for convergence, the initial conditions for the large eddy simulations were obtained from interpolating a DNS solution provided by Gavrilakis. The DNS was done using a $768 \times 127 \times 127$ grid. The current simulations were done using two different grid sizes, $129 \times 90 \times 90$ (grid A) and $257 \times 100 \times 100$ (grid B). The finer LES grid B, which gave good results, is approximately 20 percent of the total size of the DNS grid.

The convergence of the LES is determined by monitoring the time history of the total wall shear stress. For fully developed turbulent flow in a straight square duct, conservation of the mean streamwise momentum shows that the mean driving pressure gradient and the total wall shear stress are related by the following:

$$\int_{A_s} \tau_w dA = -VP_g \quad (49)$$

The surface integral is over the four side walls of the square duct, so that $A_s = 4 \times 12 \times D_H^2$. P_g is the

mean driving pressure gradient, and V is the total volume of the duct, given by $12 \times D_H^3$. Defining

$$\bar{\tau}_w = \frac{1}{A_s} \int \tau_w dA \quad \text{and} \quad P_g = \frac{dp}{dx}, \quad \text{the familiar relation}$$

between the mean pressure gradient and wall shear stress in fully developed flow in a square duct can be recovered.

$$\bar{\tau}_w = -\left(\frac{D_H}{4}\right) \frac{dp}{dx} \quad (50)$$

Figure 5 shows the time history of the total side wall shear force for the LES using grid B. The instantaneous side wall shear force level from the LES, shown as a solid line, is seen to fluctuate about a mean value, indicating that flow equilibrium has been reached in the current simulation. The time average of the computed side wall shear force is 0.001387 N. This value is in excellent agreement with the exact value of 0.001391 N, computed from equation (50) and shown as the horizontal dashed line (fig. 5). Because the time step is constant, the number of iterations shown in figure 5 is directly proportional to the time elapsed. The simulation was conducted for 218,600 time iterations, which is approximately 10 eddy turnover times (as defined in table 2).

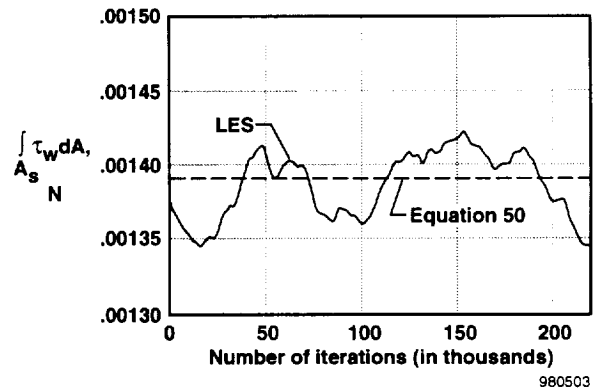


Figure 5. Convergence history for the total wall shear stress.

The parallel implementation and the results of the parallel performance studies have previously been published.^{1, 3} The code was implemented on parallel computer systems using the message-passing programming model and message-passing libraries such as Message-Passing Interface (MPI) and Parallel Virtual

Machine (PVM). The parallel speedup was found to be very good, especially for large numbers of grid points.³ Using 128 processors on the T3D computer (Cray Research, Inc.; Eagan, Minnesota), the simulation of these 10 eddy turnover times (5.5×10^{-4} sec in physical flow time) took 772 hr or approximately 1 month of central processing unit time. The same simulation would have taken approximately 150 hr on an SP2 (IBM Corporation, Austin, Texas) or T3E (Cray Research, Inc.) computer. Regardless of the computer platform used, this computational time is a large cost and shows that even with the parallel computer systems available today, turbulence simulation is still a formidable task. However, parallel CFD algorithms that can efficiently scale up with extremely large numbers of processors offer the only real hope that turbulence simulations can be done in a reasonable amount of time in the future.

The LES results shown in figures 5 to 14 are obtained using grid B and a modified Roe FDS with an ϵ_1 value of 0.03. The modification to the Roe FDS and the definition of the ϵ_1 parameter will be discussed below.

Figure 6 shows the mean streamwise velocity profile along a wall bisector. The LES solution (solid line) is compared with the DNS solution (diamond dots) supplied by Gavrilakis. The mean velocity profile in the LES was averaged both in time and space. The agreement can be seen to be very good.

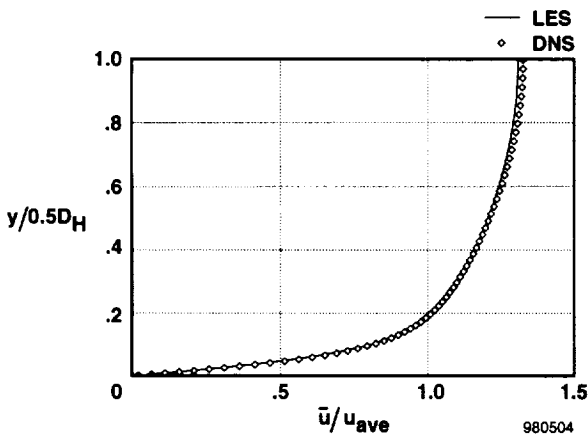


Figure 6. Mean streamwise velocity profile for fully developed turbulent flow in a square duct along the wall bisector.

Figure 7 shows the mean secondary velocity vectors from the LES. In straight ducts of noncircular cross-sections, turbulence-driven secondary flows are known to exist. These flows are different from the pressure-driven secondary flows found in curved ducts. In

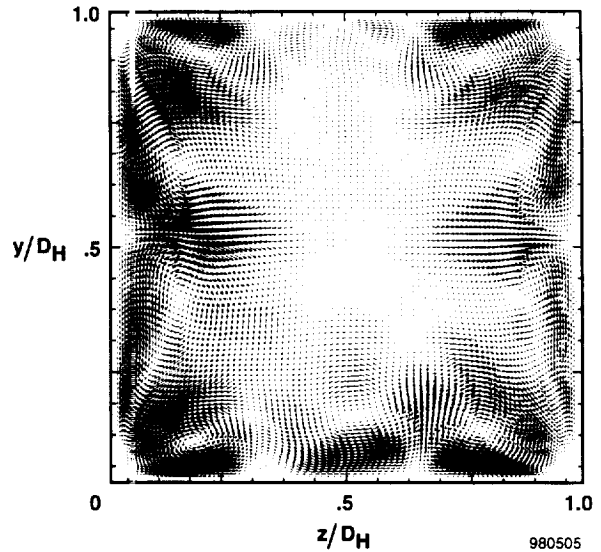


Figure 7. Mean secondary velocity vectors from LES with a $257 \times 100 \times 100$ grid.

straight square ducts, the turbulence-driven secondary flows are directed from the center of the duct toward the corners along the corner bisectors, and have been found to be produced by the anisotropy of the Reynolds stresses in the cross-sectional plane of the square duct.²² Although the magnitudes of these secondary velocities are extremely small compared to the mean average streamwise velocity (on the order of 2 percent in this simulation) these velocities have been found to be important features of this flow.

Figure 7 shows that the corner vortices produced by the secondary flows are captured in this simulation. Although some asymmetry is still evident in the plot, the overall features of the secondary flows are well-predicted by the current simulation.

To determine the accuracy of the simulation in capturing turbulence-driven secondary flows, the mean secondary velocity profiles along the lines $z/(0.5 D_H) = 0.15, 0.30, 0.50, 0.70$, and 0.80 are compared with the DNS results in figures 8 to 12. Generally, good agreement is obtained between the LES and DNS mean secondary velocity profiles.

Figures 13 and 14 show the turbulence statistics. In figure 13, the mean Reynolds stress profile along a wall bisector is compared with the DNS solution, and in figure 14, the mean turbulence intensities u_{rms} , v_{rms} , and w_{rms} are plotted. These results have been quadrant-averaged as well as averaged in space and time.

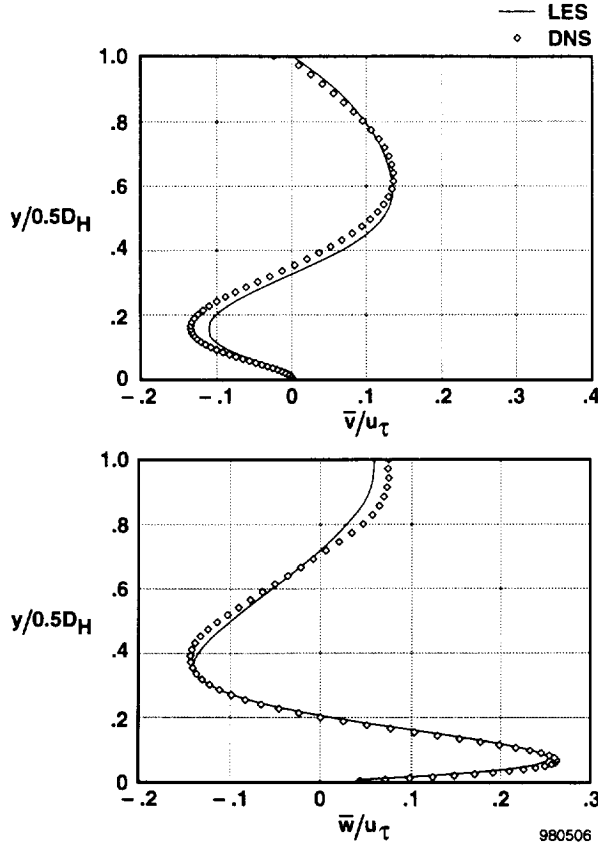


Figure 8. Mean secondary velocity profiles along $z/(0.5 D_H) = 0.15$.

Although the agreements between the LES and DNS solutions are seen to be very good for this simulation, the accuracy of the LES in capturing the turbulence velocity fluctuations was found during the turbulence simulations to be highly dependent on the numerical dissipation and the grid size used. The effects of the Roe FDS upwinding term and grid size on the computed turbulence velocity fluctuations are examined next.

Effect of the Roe Flux-Difference Splitting Term

Although the Roe FDS implemented in this code gave good results for Euler² and laminar Navier-Stokes³ test cases, the full Roe FDS term was found to be too dissipative for LES. Incorrect levels of turbulent velocity fluctuations are obtained when the normal Roe FDS term is used in turbulence simulations. This problem was solved with a simple modification to the Roe FDS algorithm. In equation (10), the inviscid fluxes normal to a cell boundary is approximated as

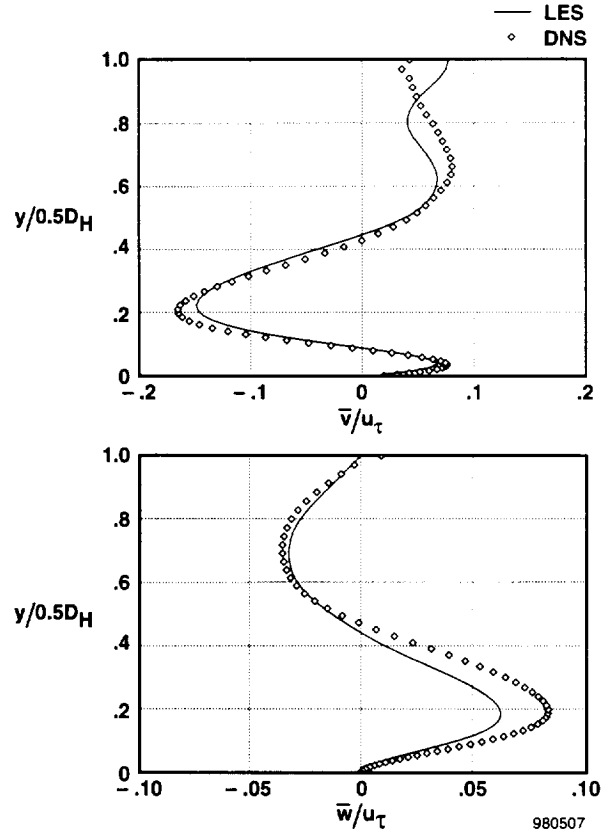


Figure 9. Mean secondary velocity profiles along $z/(0.5 D_H) = 0.3$.

$$\mathbf{f} = \frac{1}{2}(\mathbf{f}_L + \mathbf{f}_R) - \frac{1}{2}|\hat{A}|(\mathbf{U}_R - \mathbf{U}_L)$$

This approximation can be interpreted to state that the normal component of the inviscid flux at a cell boundary is the sum of the central difference of the fluxes on the left and right states, $\frac{1}{2}(\mathbf{f}_L + \mathbf{f}_R)$, plus the Roe upwinding dissipation term, $-\left\{\frac{1}{2}|\hat{A}|(\mathbf{U}_R - \mathbf{U}_L)\right\}$. If this interpretation is used, then the amount of Roe upwinding dissipation can be controlled using a multiplying factor in front of the Roe FDS term, such that

$$\mathbf{f} = \frac{1}{2}(\mathbf{f}_L + \mathbf{f}_R) - \epsilon_1 \left\{ \frac{1}{2}|\hat{A}|(\mathbf{U}_R - \mathbf{U}_L) \right\} \quad (51)$$

where ϵ_1 can range between 0 and 1. $\epsilon_1 = 0$ corresponds to central differencing only, and $\epsilon_1 = 1$ corresponds to the full Roe FDS.

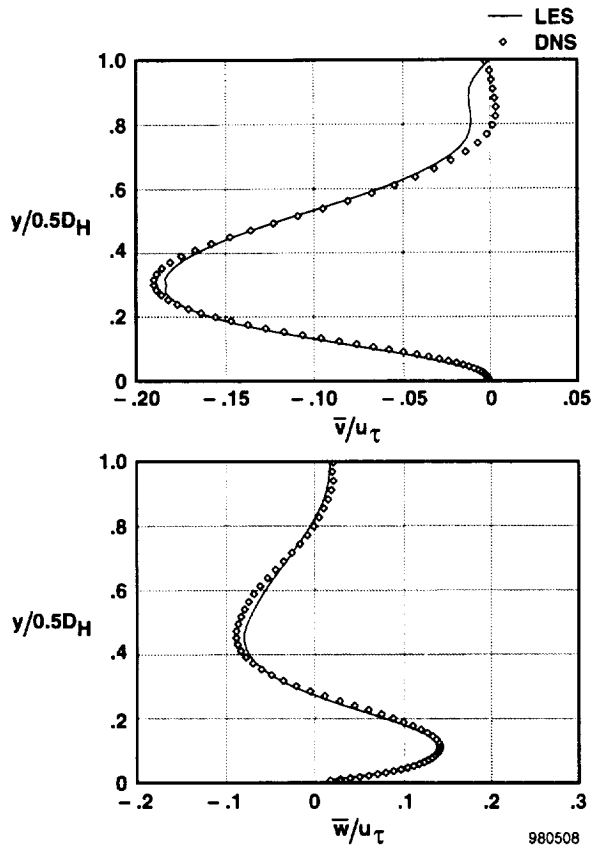


Figure 10. Mean secondary velocity profiles along $z/(0.5 D_H) = 0.5$.

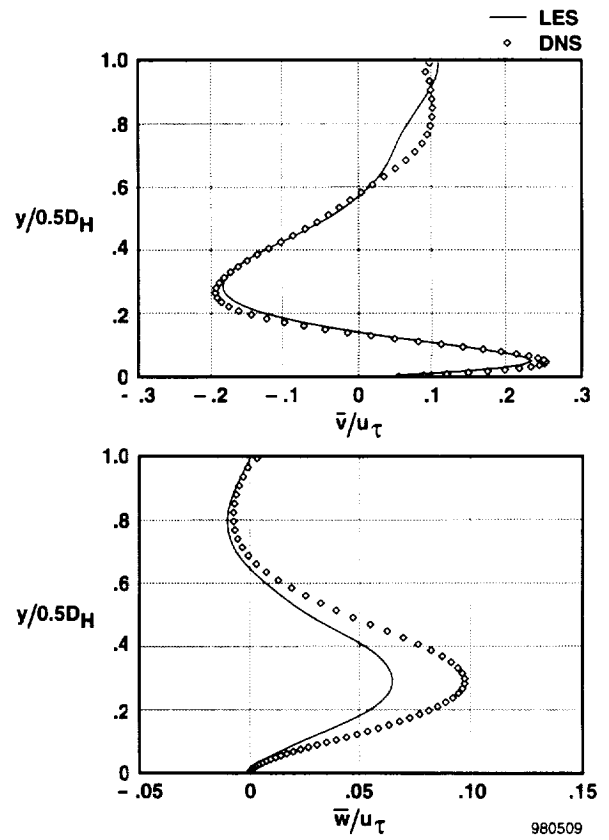


Figure 11. Mean secondary velocity profiles along $z/(0.5 D_H) = 0.7$.

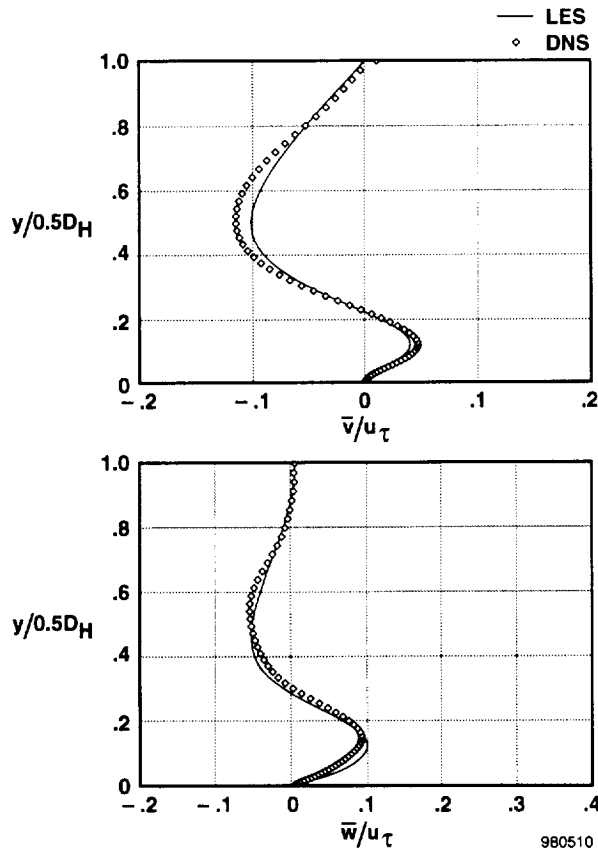


Figure 12. Mean secondary velocity profiles along $z/(0.5 D_H) = 0.8$.

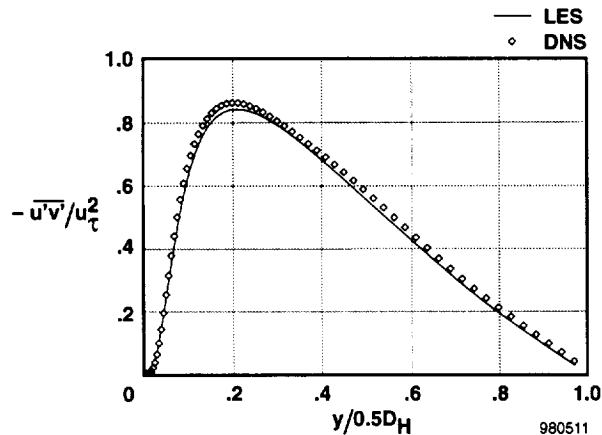


Figure 13. Mean Reynolds stress profile along the wall normal bisector.

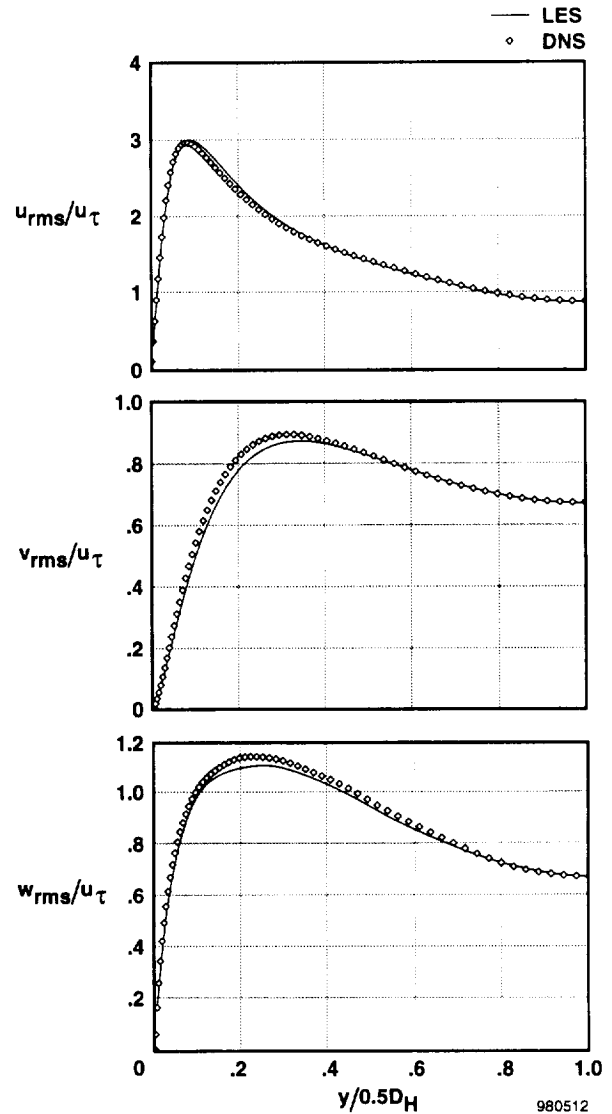


Figure 14. Turbulent intensities along the wall normal bisector.

Note that Lin et al., using the same interpretation of the Roe upwinding term as equation (51), also concluded that the normal Roe upwinding term produces too much numerical dissipation for computational aeroacoustics applications.²³ Lin et al. found that using ϵ_1 values of approximately 0.1 gave good results for acoustics computations.²³

In the LES conducted here, ϵ_1 values of less than 0.1 are needed to give good turbulence results. Omitting the Roe FDS term altogether ($\epsilon_1 = 0$) causes all calculations to be unstable, and the best turbulence solutions are

obtained using the smallest possible values of ε_1 that can still provide stable calculations. In general, the finer the grids, the smaller the minimum values of ε_1 that can be used. For a grid of $129 \times 90 \times 90$, the minimum value of ε_1 for stable calculation is 0.05, and for the $257 \times 100 \times 100$ grid, the minimum value is 0.03. The LES results presented in the preceding section were obtained using $\varepsilon_1 = 0.03$.

To study the effect of the Roe upwinding term on the turbulence simulation, LES are made for the same grid size of $129 \times 90 \times 90$ but with different values of ε_1 . Figure 15 shows the effect of Roe FDS on the mean streamwise velocity profile. Near the wall, using the full Roe FDS term produces a mean velocity gradient that is much less than both the DNS solution and the LES solution with the reduced Roe FDS. A similar effect is observed in figure 16, where the mean Reynolds stress profile obtained with $\varepsilon_1 = 1.0$ is much lower than expected.

Figure 17 also shows the excessive numerical dissipation of Roe FDS in the turbulence solution. In this figure, the solution with $\varepsilon_1 = 1.0$ gives a significantly higher level of u_{rms} and lower levels of v_{rms} and w_{rms} . Although they did not use the Roe FDS, Wang and Pletcher²⁴ reported the same problem in their LES of fully developed turbulent channel flow using an upwind CFD algorithm.

From studying the results shown in figures 15 to 17, the full Roe FDS upwinding dissipation can be seen to be detrimental to the turbulence solution, and an

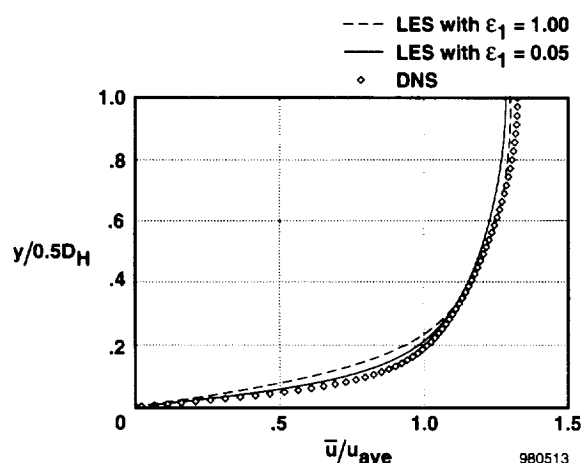


Figure 15. Effect of Roe FDS on the mean streamwise velocity profile.

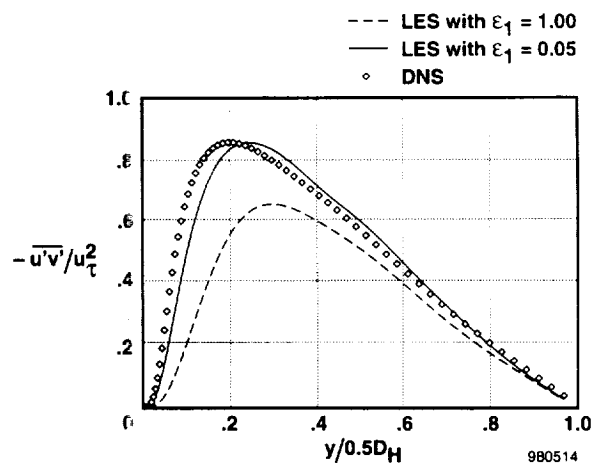


Figure 16. Effect of Roe FDS on the mean Reynolds stress profile.

improvement in the solution quality can be obtained by reducing the contribution of the Roe upwinding term. However, continuing to reduce the contribution of the Roe term until a good agreement is achieved is not possible. For a given grid size, a minimum amount of Roe FDS upwinding dissipation is required for stability. For the $129 \times 90 \times 90$ grid, the minimum ε_1 value for numerical stability is 0.05, and values smaller than this minimum will cause the calculation to be unstable. To improve the accuracy of the turbulence simulation, using a finer grid that in turn allows a smaller ε_1 value to be used is necessary. In the next section, the effect of a finer grid on the quality of the turbulence solution will be studied.

Effect of the Grid Size

The previous section showed that reducing the contribution of the Roe FDS term will improve the quality of the solution. But using a $129 \times 90 \times 90$ grid, reducing ε_1 to the minimum value of 0.05 still does not give a good agreement with the DNS solution. In this section, a finer grid with $257 \times 100 \times 100$ points is used, and the minimum value of ε_1 that can be used is lowered to 0.03.

Figure 8 shows a comparison of the mean streamwise velocity profiles. Using the finer grid in the LES produces an almost perfect agreement with the DNS solution. Figures 19 and 20 show the same improvement in the profiles of the turbulence intensities and the mean Reynolds stress, respectively.

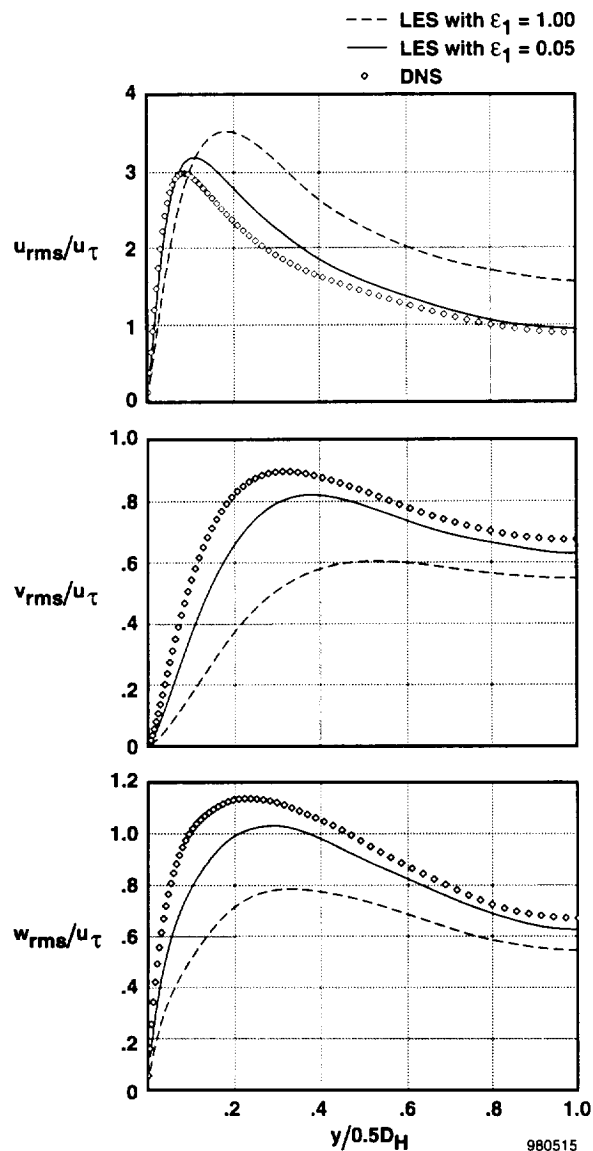


Figure 17. Effect of Roe FDS on the turbulent intensities.

Another effect of grid size can be observed by examining the mean secondary velocity vectors in a cross section of the square duct. Figure 21 shows the solution obtained using grid A and $\epsilon_1 = 0.05$. The secondary turbulence-induced velocity field is captured by the coarser grid. However, comparing this result with the finer grid B result in figure 7 shows that the corner vortices in figure 7 are somewhat smaller than those in figure 21. Figure 22 shows the streamwise-averaged

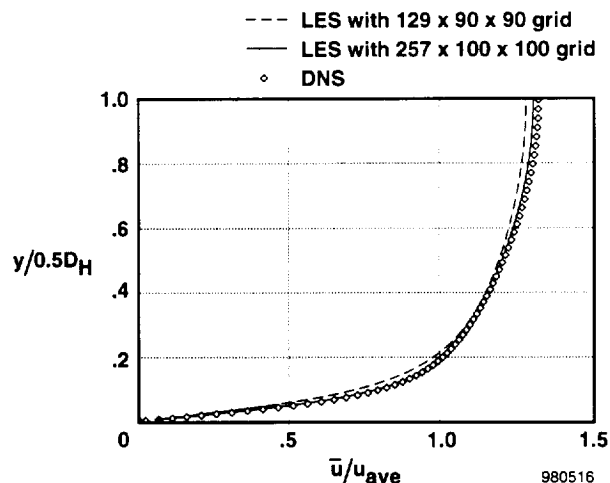


Figure 18. Effect of grid size on the mean streamwise velocity profile.

instantaneous secondary velocity vectors from the DNS solution for reference. For the DNS grid, the near wall vortical structures are even smaller than either of the LES grids. This comparison shows that the near wall turbulent structures are better resolved with finer grids.

Effect of the Subgrid-Scale Model

The previous section showed that the LES solution with the fine grid gives the best agreement with the DNS solution. Although the LES fine grid is only 20 percent of the size of the DNS grid, the LES grid density in the crossflow plane is 60 percent of the DNS crossflow plane density. As the LES grid resolution in the crossflow plane approaches the DNS resolution, the effect of the SGS model on the turbulence solution for this particular LES grid is interesting to see. An additional simulation was performed using grid B, the finer LES grid, with no SGS model. This simulation is effectively a coarse grid DNS. Figure 23 shows a comparison of the mean streamwise velocity profiles. The use of the SGS model makes essentially no difference in the mean streamwise velocity solution. Small differences are also observed in the turbulent velocity fluctuations shown in figure 24. Figure 25 shows a comparison of the mean Reynolds stress profiles. The biggest difference is at the peak of the Reynolds stress profile, where the LES solution with no SGS model predicts a slightly higher peak than the DNS, and the LES solution with the SGS model predicts

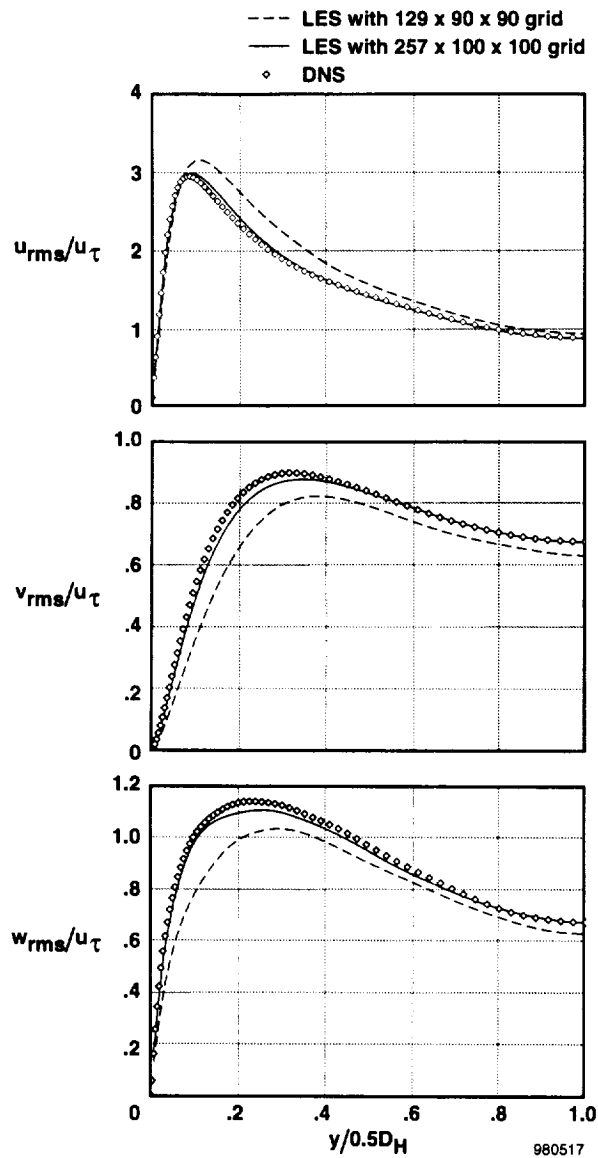


Figure 19. Effect of grid size on the turbulent intensities.

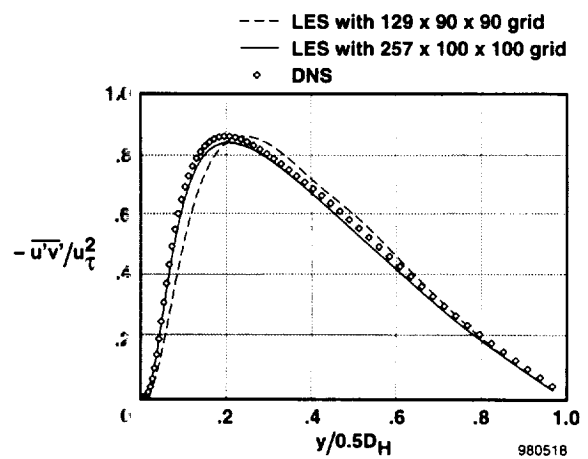


Figure 20. Effect of grid size on the mean Reynolds stress profile.

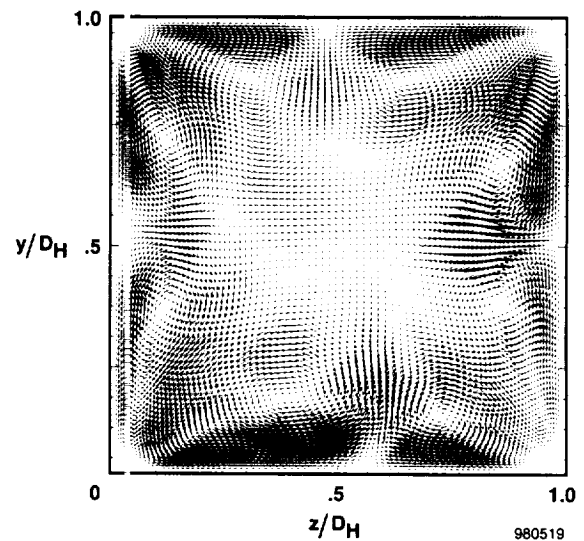


Figure 21. Mean secondary velocity vectors from LES with a 129 x 90 x 90 grid.

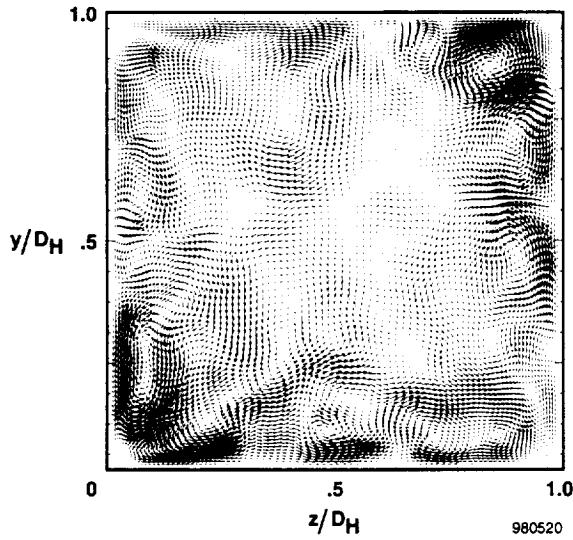


Figure 22. Streamwise-averaged secondary velocity vectors from the DNS solution.

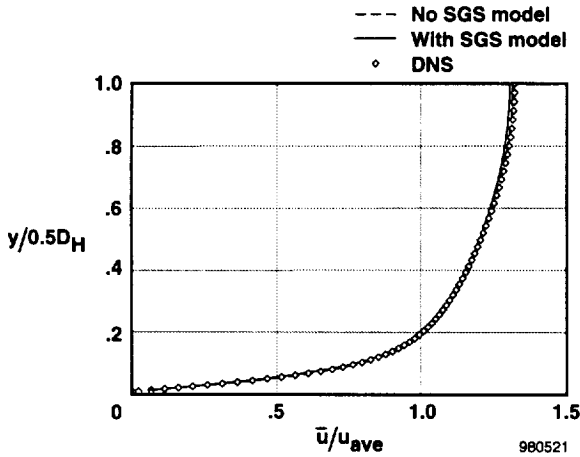


Figure 23. Effect of the SGS model on the mean streamwise velocity profile.

a slightly lower peak than the DNS. Note that the LES turbulence statistics were only computed for the resolved velocity field. As a result, the LES Reynolds stress profile with the SGS model should be lower than the DNS solution because the SGS contribution was not included.

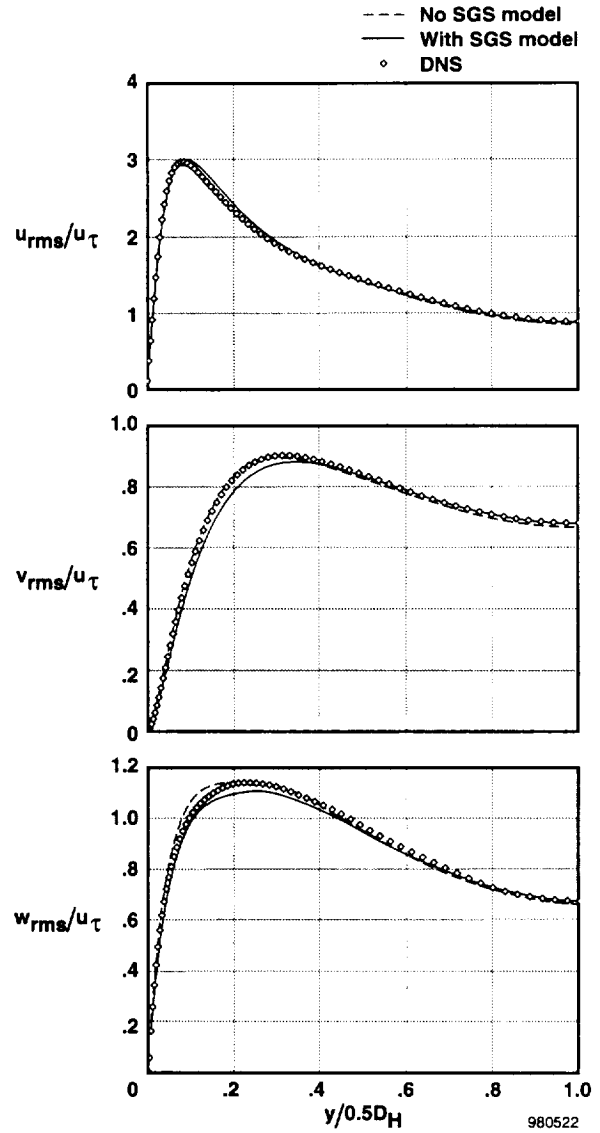


Figure 24. Effect of the SGS model on the turbulent intensities.

These results indicate that an SGS model is not needed for an accurate simulation of this test case. As discussed earlier, the grid resolution in the near wall region has to be very fine to resolve the small energy-producing structures there. The fine grid LES conducted here has effectively approached the DNS in the near wall limit.

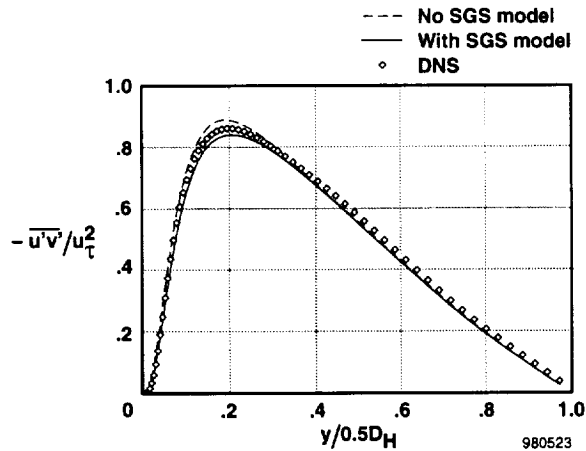


Figure 25. Effect of the SGS model on the mean Reynolds stress profile.

Conclusion

A new, parallel, finite-volume computational fluid dynamics algorithm was developed for large-eddy simulation (LES) of turbulent flows using parallel computer systems. Major components of the algorithm included piecewise linear least-square reconstruction of the unknown variables, trilinear finite-element interpolation for the spatial coordinates, Roe flux-difference splitting (FDS), and second-order MacCormack explicit time marching. The parallel implementation was accomplished using the message-passing programming model.

For the first time, a parallel, unstructured, finite-volume numerical algorithm was used for LES of turbulent flow in a square duct, and several conclusions have been drawn regarding the accuracy and efficiency of this numerical algorithm. Comparison with the direct numerical simulation (DNS) solution showed that the standard Roe FDS upwind dissipation adversely affects the accuracy of the turbulence simulations. A modification to the standard Roe FDS method was proposed in which the inviscid flux is computed as the arithmetic average of the right and left fluxes plus the product of the Roe FDS dissipation term and a reduction factor. For accurate turbulence simulations, only 3–5 percent of the normal Roe FDS dissipation was found to be needed.

The finer, $257 \times 100 \times 100$ LES grid required less Roe FDS upwind dissipation for stability and produced a more accurate solution than the $129 \times 90 \times 90$ LES grid. The near wall vortical structures were better simulated

by the finer grid LES, and the effect of the subgrid-scale model on the accuracy of the results was found to be small for the fine grid LES, which is nearly as fine as the DNS grid in the near wall region.

Acknowledgments

The author would like to thank Dr. Spyros Gavrilakis of the Ecole Polytechnique Federale de Lausanne, Switzerland, for graciously providing the DNS solution used in this study. The author also thanks Dr. Tom Lund of the University of Texas at Arlington for his helpful advice on the topic of LES.

References

- ¹Bui, Trong Tri, *A Parallel Finite Volume Algorithm for Large-Eddy Simulation of Turbulent Flows*, Ph. D. thesis, Department of Aeronautics and Astronautics, Stanford University, California, Mar. 1998.
- ²Bui, Trong T. and Reda R. Mankbadi, "Direct Numerical Simulation of Acoustic Waves Interacting With a Shock Wave in a Quasi-1D Convergent-Divergent Nozzle Using an Unstructured Finite Volume Algorithm," FED-Vol. 219, Computational Aeroacoustics, ASME 1995.
- ³Bui, Trong T., "Numerical Simulation of 3-D Low Mach Number Viscous Duct Flows Using an Explicit Method on Massively Parallel Computer Systems," AIAA-97-0335, Jan. 1997.
- ⁴Barth, Timothy J., "Recent Developments in High Order K-Exact Reconstruction on Unstructured Meshes," AIAA-93-0668, Jan. 1993.
- ⁵Coirier, William John, "An Adaptively-Refined, Cartesian, Cell-Based Scheme for the Euler and Navier-Stokes Equations," NASA TM-106754, 1994.
- ⁶Hughes Thomas J. R., *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1987.
- ⁷Greenberg, Michael D., *Foundations of Applied Mathematics*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1978.
- ⁸Moin, P., K. Squires, W. Cabot, and S. Lee, "A Dynamic Subgrid-Scale Model for Compressible Turbulence and Scalar Transport," *Physics of Fluids A*, vol. 3, Nov 1991, pp. 2746–2757.